CLAIMS

We claim:

1. A method for verifying lockstep operation, the method comprising:

monitoring interface signals;

detecting output of a modeled lockstep block;

comparing the detected output with an expected output for the lockstep block relative to a current modeled machine state; and

flagging a lockstep block error if the detected output does not match the expected output.

2. The method of claim 1, wherein monitoring interface signals comprises monitoring signals on a point-to-point interface of a register transfer language simulator.

3. The method of claim 1, wherein monitoring interface signals comprises monitoring for output signals from modeled processor cores and the modeled lockstep block.

4. The method of claim 3, wherein monitoring interface signals comprises detecting an error signal output by one of the modeled processor cores.

5. The method of claim 4, further comprising transitioning a state machine model into a core-disabled mode.

6. The method of claim 5, further comprising examining an output error signal of the modeled lockstep block to determine when the output error signal was fired.

7. The method of claim 6, wherein comparing the detected output with an expected output comprises determining whether the output error signal was fired at a time when that signal was expected and wherein flagging a lockstep block error comprises flagging a lockstep block error if the output error signal was not fired when expected.

8. The method of claim 5, further comprising comparing data values for a healthy core with the detected output.

9. The method of claim 8, wherein flagging a lockstep error comprises flagging a lockstep error if the data values do not match expected data values.

10. The method of claim 3, further comprising capturing output values of at least two modeled processor cores into a state machine model and comparing the captured output values.

11. The method of claim 10, further comprising transitioning the state machine model into a difference-detected mode if the compared values are different.

12. The method of claim 11, further comprising examining a fatal error signal from the modeled lockstep block to determine when that signal was fired.

13. The method of claim 12, wherein comparing the detected output with an expected output comprises determining whether the output error signal was fired at a time when that signal was expected and wherein flagging a lockstep error comprises flagging a lockstep block error if the output error signal was not fired at the expected time.

14. A system for verifying lockstep operation, the system comprising:

means for monitoring interface signals output by modeled processor cores and a modeled lockstep block;

means for determining an expected output from the modeled lockstep block relative to the monitored output from the modeled processor cores;

means for comparing output from the modeled lockstep block with the expected output; and

means for flagging a lockstep block error if the detected output does not match the expected output.

15. The system of claim 14, wherein the means for monitoring interface signals comprise means for monitoring signals on a point-to-point interface of a register transfer language simulator.

16. The system of claim 14, wherein the means for determining an expected output comprise a data structure that relates processor core outputs with expected lockstep block outputs.

17. The system of claim 14, further comprising means for transitioning a state machine model into one of a core-disabled mode and a difference-detected mode.

18. The system of claim 14, further comprising means for examining an output error signal of the modeled lockstep block to determine when the output error signal was fired.

19. The system of claim 14, further comprising means for comparing data values for a healthy core with the output of the modeled lockstep block.

20. The system of claim 14, further comprising means for comparing output values of the modeled processor cores.

21. The system of claim 14, further comprising means for examining a fatal error signal from the modeled lockstep block to determine when that signal was fired.

22. A lockstep block checker stored on a computer-readable medium, the checker comprising:

logic configured to monitor a point-to-point interface for interface signals output by modeled processor cores and a modeled lockstep block;

logic configured to determine an expected output from the modeled lockstep block; and

logic configured to compare output from the modeled lockstep block with the expected output and flag a lockstep block error if the detected output does not match the expected output.

23. The checker of claim 22, further comprising logic configured to transition a state machine model into one of a core-disabled mode and a difference-detected mode based upon modeled processor core output.

24. The checker of claim 22, further comprising logic configured to examine an output error signal of the modeled lockstep block to determine when the output error signal was fired.

25. The checker of claim 22, further comprising logic configured to compare data values for a healthy core with the output of the modeled lockstep block..

26. The checker of claim 22, further comprising logic configured to compare output values of the modeled processor cores.

27. The checker of claim 22, further comprising logic configured to examine a fatal error signal from the modeled lockstep block to determine when that signal was fired.

28. A computer system, comprising:

a processing device; and

memory including a lockstep block checker and a register transfer language simulator that models processor cores and a lockstep block, wherein the checker is configured to monitor an interface of the simulator for interface signals output by the modeled processor cores and the modeled lockstep block, determine an expected output from the modeled lockstep block, compare output from the modeled lockstep block with the expected output, and flag a lockstep block error if the detected output does not match the expected output.

29. The system of claim 28, wherein the lockstep block checker comprises a state machine model and is further configured to transition the state machine model into one of a core-disabled mode and a difference-detected mode based upon modeled processor core output.

30. The system of claim 28, wherein the lockstep block checker is further configured to examine an output error signal of the modeled lockstep block to determine when the output error signal was fired.

31. The system of claim 28, wherein the lockstep block checker is further configured to compare data values for a healthy core with the output of the modeled lockstep block.

32. The system of claim 28, wherein the lockstep block checker is further configured to compare output values of the modeled processor cores.

33. The system of claim 28, wherein the lockstep block checker is further configured to examine a fatal error signal from the modeled lockstep block to determine when that signal was fired.